Soft Computing manuscript No.
(will be inserted by the editor)

Improving the Multiobjective Evolutionary Algorithm Based on Decomposition with New Penalty Schemes

Shengxiang Yang · Shouyong Jiang · Yong Jiang

Received: September 23, 2015; Revised: December 9, 2015; Accepted: January 26, 2016

Abstract It has been increasingly reported that the multiobjective optimization evolutionary algorithm based on decomposition (MOEA/D) is promising for handling multiobjective optimization problems (MOPs). MOEA/D employs scalarizing functions to convert an MOP into a number of single-objective subproblems. Among them, penalty boundary intersection (PBI) is one of the most popular decomposition approaches and has been widely adopted for dealing with MOPs. However, the original PBI uses a constant penalty value for all subproblems and has difficulties in achieving a good distribution and coverage of the Pareto front for some problems. In this paper, we investigate the influence of the penalty factor on PBI, and suggest two new penalty schemes, i.e., adaptive penalty scheme (APS) and subproblem-based penalty scheme (SPS), to enhance the spread of Pareto optimal solutions. The new penalty schemes are examined on several complex MOPs, showing that PBI with the use of them is able to provide a better approximation of the Pareto front than the original one. The SPS is further integrated into two recently developed MOEA/D variants to help balance the population diversity and convergence. Experimental results show that it can significantly enhance the algorithm's performance.

Shengxiang Yang
School of Mathematics and Statistics, Nanjing University of Information Science and Technology, Nanjing 210044, China; and Centre for Computational Intelligence (CCI), School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K. E-mail: syang@dmu.ac.uk

Shouyong Jiang
Centre for Computational Intelligence (CCI), School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K. E-mail: shouyong.jiang@email.dmu.ac.uk

Yong Jiang
School of Mathematics and Statistics, Nanjing University of Information Science and Technology, Nanjing 210044, China. E-mail: jiang@nuist.edu.cn

Keywords Multiobjective evolutionary algorithm · decomposition · penalty boundary intersection · adaptive penalty scheme · subproblem-based penalty scheme

1 Introduction

Many real-world optimization problems, such as water resource management (Reed et al., 2013), design optimization (Deb and Jain, 2014; Yang and Deb, 2013), and land use management (Chikumbo, 2012; Masoomi et al., 2013), often have multiple objectives that conflict with each other. Without lose of generality, these kinds of multiobjective optimization problems (MOPs) can be described as follows:

$$\begin{aligned} &min\ \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_M(\mathbf{x}))^T \\ &s.t.\ \begin{cases} h_i(\mathbf{x}) = 0, & i = 1, ..., n_h \\ g_i(\mathbf{x}) \geq 0, & i = 1, ..., n_g \\ \mathbf{x} \in \Omega_x \end{cases} \end{aligned} \tag{1}$$

where $\Omega_x \subseteq R^n$ is the decision space, n_h and n_g are the number of equalities and inequalities, respectively, M is the number of objectives, and $\mathbf{F}(\mathbf{x})$: $\Omega_x \rightarrow R^M$ is the objective function vector of the solution $\mathbf{x}$. Due to the nature of conflict among objectives, there does not exist a single optimal solution that can minimize all the objectives simultaneously. As a consequence, the optimization goal of MOPs is to obtain a set of solutions with a good trade-off among the objectives. The set of trade-off solutions is known as the Pareto-optimal set (POS), and its image in the objective space is called the Pareto-optimal front (POF).

Multiobjective evolutionary algorithms (MOEAs) are an important class of methods devoted to solving MOPs. They employ a population of candidate members and evolve them collaboratively, resulting in a number of candidate solutions in a single run, without the use of any gradient information of MOPs. These characteristics make MOEAs very suitable

for handling MOPs. After several decades of effort to the field of evolutionary computation, a huge number of MOEAs are available to date. They can be classified into three main groups: Pareto-based approaches (Deb et al., 2002; Knowles and Corne, 1999; Zitzler et al., 2002; Deb and Jain, 2014), indicator-based approaches (Bader and Zitzler, 2011; Beume et al., 2007; Ishibuchi et al., 2010; Zitzler and Kunzli, 2004), and decomposition-based approaches (Zhang and Li, 2007; Li and Zhang, 2009; Asafuddoula et al., 2015). The MOEA based on decomposition (MOEA/D) is the most well-known representative of decomposition-based approaches. It uses scalarizing functions or decomposition approaches to decompose an MOP into a number of single-objective subproblems and optimize them in a co-evolutionary manner. In MOEA/D, weight vectors or search directions involved in the scalarizing functions implicitly manage population diversity, and the concept of neighbourhood is introduced to co-evolve solutions of neighbouring subproblems. This way, MOEA/D can quickly approximate the Pareto front and provide a set of diverse solutions. Recently, various versions of MOEA/D have been proposed in the literature (Zhang and Li, 2007; Li and Zhang, 2009; Asafuddoula et al., 2015; Li et al., 2015a; Jiang and Yang, 2015), and the idea of decomposition has been exploited in a number of studies (Deb and Jain, 2014; Li et al., 2014b,c, 2015a,b; Liu et al., 2014; Yuan et al., 2015).

In MOEA/D, there are three widely-used scalarizing functions, i.e., weighted sum, weighted Tchebycheff, and penalty boundary intersection (PBI), to aggregate M different objectives (Zhang and Li, 2007). Compared with PBI, the weighted sum and weighted Tchebycheff are easier to implement and less computationally expensive. A recent study (Ishibuchi et al., 2013) reported that the weighted sum shows better search performance than the weighted Tchebycheff in many-objective problems. However, the weighted sum is not effective to approximate problems with the entire concave Pareto front (Zhang and Li, 2007). The weighted Tchebycheff approach has received intensive research interest due to its ability to approximate both convex and concave POFs. Despite its great success for solving standard benchmark problems like ZDT (Zitzler et al., 2000) or DTLZ (Deb et al, 2005), some recent investigations have revealed that this approach has difficulties in uniformly distributing solutions on boundary regions of the POF for complex problems (Qi et al., 2014; Jiang and Yang, 2015). On the other hand, the PBI approach gains a firm foothold in MOEA/D because it can provide a more uniform distribution of POF than the weighted Tchebycheff for three- and higher-dimensional problems (Li et al., 2015a; Zhang and Li, 2007; Deb and Jain, 2014; Gomez and Coello Coello, 2015).

It is not very surprising that PBI has obtained great success for multiobjective and many-objective problems since its introduction. Probably due to the lack of complicated POF characteristics in the well-known ZDT (Zitzler et al., 2000), DTLZ (Deb et al, 2005) and WFG (Huband et al., 2006) test suites in the field of multiobjective optimization, PBI has been rarely or even not deeply examined on problems with complicated POFs. However, real-world MOPs often have irregular POF geometries (Deb et al., 2000; Osycza and Kundu, 1995; Wang et al., 2013), where the shape can be extremely convex and/or concave. Note that, there are also other kinds of irregularities, such as degenerate and disconnected POFs (Cheng et al., 2015; Jain and Deb, 2013; Giagkiozis et al., 2014), which may be harder than extremely-shaped POFs. As a starting point, we just consider irregular extremely-shaped POFs in this paper. While it has been repeatedly reported that PBI works well on regular problems, one may wonder whether PBI can continue its success on the real-life or artificial problems with complicated POFs. If not, the applicability of PBI (Zhang and Li, 2007; Al Moubayed et al., 2013; Pavelski et al., 2014; Jia et al., 2011) and its extension to many-objective optimization (Cheng et al., 2016; Deb and Jain, 2014; Gomez and Coello Coello, 2015; Li et al., 2014a, 2015a; Mohammadi et al., 2014; Yuan et al., 2014; Mendez and Coello Coello, 2015; Gomez and Coello Coello, 2015), which has become a recent hot topic, could be questioned.

The performance of PBI is largely determined by its penalty factor, which controls the balance between convergence and diversity. A small penalty favours convergence whereas a large one stresses diversity. The penalty value was commonly set to 5.0 in most studies, e.g., (Deb and Jain, 2014; Zhang and Li, 2007; Li et al., 2015a). However, in (Ishibuchi et al., 2015), PBI with a penalty value of 0.1 was reported to outperform that with a penalty value of 5.0 on multiobjective knapsack problems. This means, the optimal value for the penalty factor may vary dramatically from problems to problems. Also, in the case of small penalty values, if the distance from the obtained ideal point to boundary solutions is significantly larger than that to intermediate solutions, boundary solutions are likely to be abandoned during the search because of the poor convergence measured by the distance. In the case of large penalty values, due to overemphasis of diversity, population individuals need more computational resources to explore the search space before converging to the POF. Besides, Sato (2014) argued that if the obtained ideal point is far from the true ideal point (probably due to the loss of boundary solutions), PBI will face difficulties to approximate the entire POF. Therefore, the author suggested an inverted version of PBI to ease this problem. Nevertheless, the inverted PBI fails to consider the influence of the penalty factor, and may still get stuck into the difficulty of tuning this parameter.

In this paper, we first study the influence of the penalty factor on the search performance of PBI, followed by suggesting two new penalty schemes, i.e., adap-

tive penalty scheme (APS) and subproblem-based penalty scheme (SPS). APS adaptively assigns all subproblems the same penalty value by taking into consideration different requirements at different search stages, whereas SPS specifies distinct penalty values for different subproblems at an attempt to enhance the spread of POF. These schemes are examined on several complex problems and experimental results demonstrate their promise for improving PBI's performance. Besides, SPS is further integrated into some newly-developed MOEA/D variants, showing that it significantly improves the performance of these algorithms.

The rest of this paper is outlined as follows. Section 2 briefly reviews related work on MOEA/D. Section 3 first presents an investigation into the influence of the penalty factor on PBI and then introduces the two penalty schemes proposed for PBI. In Section 4, the experimental study is conducted on the proposed penalty schemes. Section 5 concludes this paper and discusses some future work.

2 Related Work

2.1 The PBI Approach

Decomposition approaches play a key role in converting an MOP into a number of scalar optimization subproblems in decomposition-based MOEAs. There are three popular decomposition approaches, such as the weighted sum, weighted Tchebycheff and PBI. The weighted sum approach favours convex problems and fails to approximate problems with a concave POF, whereas the weighted Tchebycheff approach can handle both convex and concave problems. The PBI approach has its advantages in obtaining a good distribution of solutions in the objective space (Zhang and Li, 2007) and handling many-objective problems (Deb and Jain, 2014; Ishibuchi et al., 2015), but its performance is very sensitive to the setting of the penalty factor (Ishibuchi et al., 2015). The scalar optimization problem of PBI is defined as follows:

$$\begin{aligned} &\min\ g^{pbi}(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = d_1 + \theta d_2 \\ &s.t.\ \mathbf{x} \in \Omega_x \end{aligned} \tag{2}$$

where

$$d_1 = \frac{\|(\mathbf{F}(\mathbf{x}) - \mathbf{z}^*)^T \mathbf{w}\|}{\|\mathbf{w}\|} \tag{3}$$

$$d_2 = \|\mathbf{F}(\mathbf{x}) - (\mathbf{z}^* + d_1 \frac{\mathbf{w}}{\|\mathbf{w}\|})\| \tag{4}$$

Ω_x is the search space and $z^* = (z_1^*, \cdots, z_M^*)$ is the ideal point in the objective space for which the ith component can be computed by $z_i^* = \min_{\mathbf{x}\in\Omega_x} f_i(\mathbf{x})$, and θ is a user-defined penalty factor. d_1 and d_2 are the length of the projection of vector $(\mathbf{F}(\mathbf{x}) - \mathbf{z}^*)$ on the weight vector $\mathbf{w}$ and the perpendicular distance from $\mathbf{F}(\mathbf{x})$ to $\mathbf{w}$, respectively. Fig. 1 offers a brief illustration of the PBI approach. It is not difficult to see that θ takes the responsibility for balancing convergence (measured by d_1) and diversity (measured by d_2), and the PBI approach drives the search toward the obtained ideal point z^* by minimizing g^{pbi}.

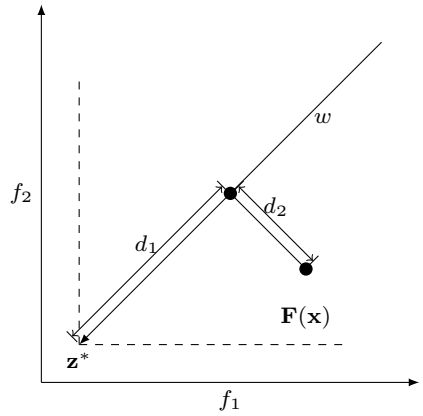
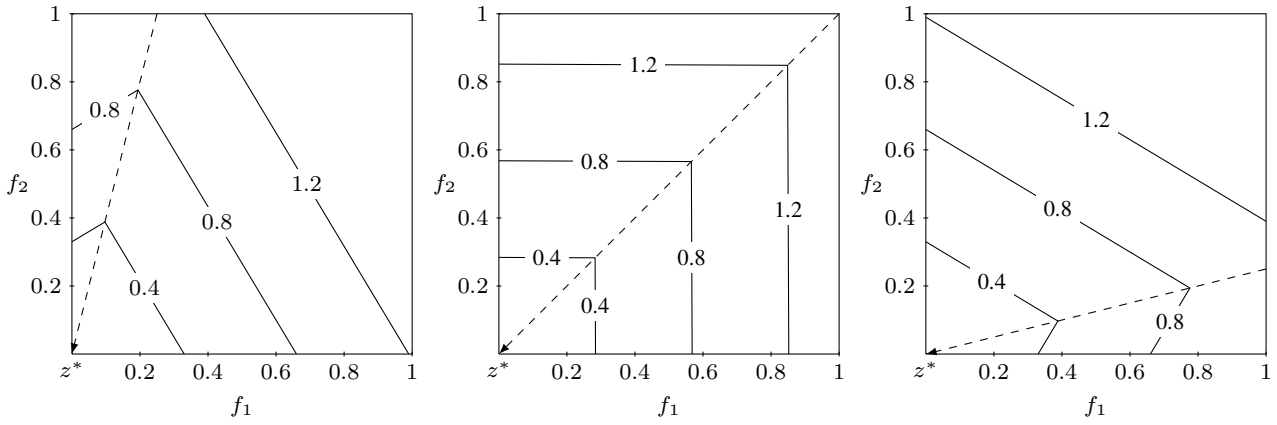


Fig. 1 Illustration of PBI.

Figure 2 shows the contour lines of PBI with $\theta = 1.0$ on three different two-dimensional weight vectors, where different weight vectors specify distinct promising search regions in the objective space. This means each weight vector can direct the search toward its preferred POF segment, and by employing a number of well-spread weight vectors, the whole POF can be approximated. To illustrate the influence of the penalty factor θ, we also plot the contour lines of PBI on $\mathbf{w} = (0.5, 0.5)^T$ for $\theta = 0.0$, 1.0 and 2.0 in Fig. 3. The figure indicates that different values of θ lead to different search behaviours and a large value of θ narrows efficient search regions and promotes diversity.

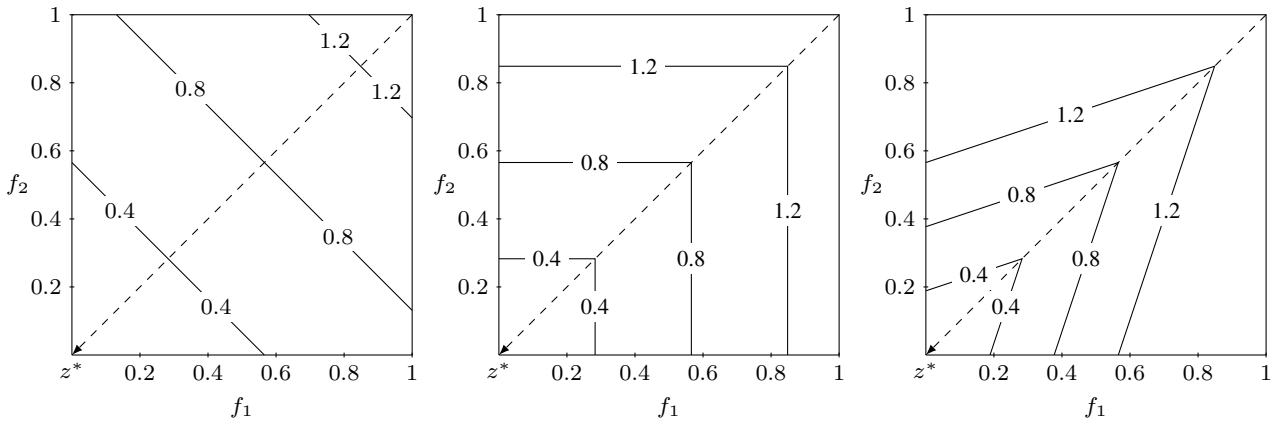
2.2 MOEA/D Algorithm

MOEA/D employs a set of predefined weight vectors that uniformly partition the entire objective space to specify a number of search directions, and defines a single-objective problem or a multiobjective subproblem by decomposition approaches for each search direction. For each search direction, MOEA/D also specifies T closest neighbours beforehand, which helps to efficiently solve the associated single-objective problem in a collaborative manner. During the course of search, mating selection and replacement are considered among solutions associated with the T neighbouring search directions. MOEA/D is a steady-state algorithm and updates solutions one by one, so it approximates the true POF quickly. Algorithm 1 presents a brief description of the original MOEA/D with the PBI (Zhang and Li, 2007). For more details, the interested readers are referred to (Zhang and Li, 2007).



(a) $w = (0.2, 0.8)^T$ (b) $w = (0.5, 0.5)^T$ (c) $w = (0.8, 0.2)^T$

Fig. 2 Contour lines of PBI with $\theta = 1.0$.



(a) $\theta = 0.0$ (b) $\theta = 1.0$ (c) $\theta = 2.0$

Fig. 3 Contour lines of PBI with $w = (0.5, 0.5)^T$.

Algorithm 1 MOEA/D-PBI

1: **Input**:
   - $MaxIteration$: the stopping criterion
   - N: the number of subproblems considered in MOEA/D
   - T: the neighbourhood size
2: **Output**: approximated Pareto-optimal set
3: **Initialization:** Generate a uniform spread of N weight vectors: $w^1, \ldots, w^N$ and then compute the T closest weight vectors to each weight vector by the Euclidean distance. For each $i = 1, \ldots, N$, set $B(i) = \{i_1, \ldots, i_T\}$ where $w^{i_1}, \ldots, w^{i_T}$ are the T closest weight vectors to w^i
4: Generate an initial population $P = \{x^1, \ldots, x^N\}$ by uniformly randomly sampling from the decision space
5: **while** $gen := 1$ to $MaxIteration$ **do**
6: **for** $i := 1$ to N **do**
7: Randomly select two indexes r_1 and r_2 from $B(i)$
8: Apply genetic operators on individuals r_1, r_2 to produce a new solution y
9: If y is better than any individual x^j in $B(i)$ (i.e., $g^{pbi}(y|w^j, z^*) \leq g^{pbi}(x^j|w^j, z^*)$), then x^j is replaced by y
10: **end for**
11: **end while**
12: Output P

3 Proposed Penalty Schemes for MOEA/D with PBI

3.1 Influence of the Penalty Factor

As mentioned earlier, θ is a key factor for balancing convergence and diversity in PBI. A small value of θ favours convergence whereas a large one is beneficial for diversity. Figures 4 and 5 show the influence of two different θ values on convergence and diversity, where the bold curve is the true POF and points A to G are scattered POF points.

In Fig. 4, θ is set to 1.0, and its influence is illustrated by two adjacent boundary weight vectors, i.e., $w_1 = (0.1, 0.9)^T$ and $w_2 = (0.2, 0.8)^T$. Intuitively, B and C are ideal optimal points for subproblems associated with w_1 and w_2, respectively. However, the solution B of the subproblem associated with w_1 will be replaced with C since $g^{pbi}(C|w_1, z^*) = 0.60$ is smaller than $g^{pbi}(B|w_1, z^*) = 0.75$. This means, due to insufficient penalty, POF points far away from the obtained ideal point z^* are replaced by those close to z^*, thus the diversity of solutions declines. Even

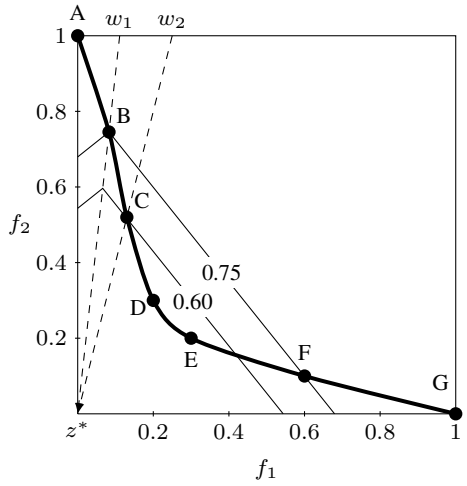


Fig. 4 Illustration of insufficient penalty for weight vectors where $w_1 = (0.1, 0.9)^T$, $w_2 = (0.2, 0.8)^T$ and $\theta = 1.0$ is used in PBI.

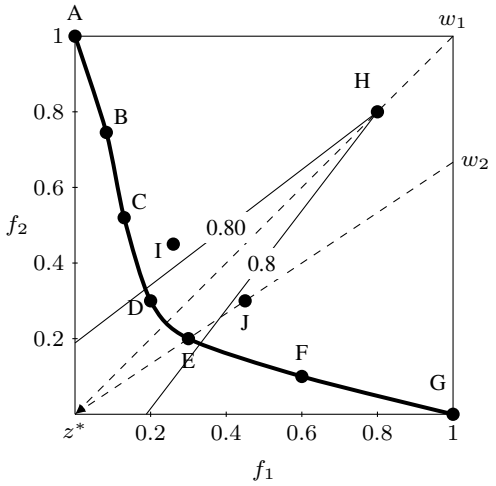


Fig. 5 Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)^T$, $w_2 = (0.4, 0.6)^T$ and $\theta = 5.0$ is used in PBI.

worse, for extremely-convex problems that have a sharp peak and long tail (Jiang and Yang, 2015), the above impact is more vital, and boundary points on the POF cannot be approximated so as to decrease the spread of the POF.

In Fig. 5, $w_1 = (0.5, 0.5)^T$ and $w_2 = (0.4, 0.6)^T$ are two adjacent intermediate weight vectors, and a large value of $\theta = 5.0$ is used in PBI. For w_1, PBI is expected to drive the search toward the POF segment between D and E. At the early stage of search, the subproblems associated with w_1 and w_2 find their best solution H and J, respectively. Clearly, J is much closer than H to the expected POF segment. But, for the subproblem associated with w_1, its current solution H will not be replaced by J as J gives a larger g^{pbi} than H. This indicates, due to excessive penalty, solutions close to weight vectors but far away from the POF may be preferred over those close to the POF but far away from the weight vectors. This inevitably spoils PBI's convergence performance and slows down the whole search process.

3.2 Adaptive Penalty Scheme (APS)

Facing the above-mentioned difficulties, one may naturally turn to adaptively adjusting the magnitude of penalty at different search stages. At the early stage, the main optimization goal is to drive the search toward the POF as fast as possible, so a small value of θ is helpful for fast convergence. In contrast, at the late stage, the obtained solutions are required to be diverse and spread so that they will not miss any part of the entire POF. Thus, the late search stage focuses on diversity, and a large value of θ is expected to diversify the solutions.

In this paper, we propose to use the APS to balance the population convergence and diversity at different stages, which is described as follows:

$$\theta = \theta_{min} + (\theta_{max} - \theta_{min})\frac{t}{maxIteration} \tag{5}$$

where t is the iteration counter, $maxIteration$ is the maximum number of iterations, θ_{min} and θ_{max} are the lower and upper bounds of θ, respectively. Here, these two parameters are suggested to be $\theta_{min} = 1.0$ and $\theta_{max} = 10.0$, which ensures $\theta = 5$ recommended by Zhang and Li (2007) is in the interval $[\theta_{min}, \theta_{max}]$.

Note that, the APS used in this paper is linear and serves as an example for validating the promise of this kind of method to enhance PBI. In principle, any proper nonlinear APS can fulfil this purpose.

3.3 Subproblem-based Penalty Scheme (SPS)

Another alternative way to reduce the drawback of PBI is to specify independently a penalty value for each subproblem. As explained in Section 3.1, for convex problems, boundary solutions are much further than intermediate solutions from the obtained ideal point, so boundary subproblems should be given strict penalties on diversity measure to avoid unreasonable replacements. For concave problems, although there is no much difference between boundary solutions and intermediate solutions in terms of their distance to the obtained ideal point, strict penalty on boundary solutions might also help maintain diversity.

The idea behind the SPS is simple, but it is not easy to assign N different penalty values for N subproblems. Also, such assignments may be computationally expensive. Having realized that each subproblem is mainly determined by its weight vector and there are distinct differences between boundary and intermediate weight vectors, we elaborate the following SPS for PBI:

$$\theta_i = e^{\alpha\beta_i} \tag{6}$$

$$\beta_i = \max_{1\le j\le M} w_i^j - \min_{1\le j\le M} w_i^j \tag{7}$$

where θ_i denotes the penalty value for a weight vector w_i, and β_i is the difference between the maximum component and the minimum component in w_i. Since $\sum_{j=1}^{M} w_i^j = 1$ and $w_i^j \geq 0$ for any $1 \leq j \leq M$, β_i lies in [0,1]. For intermediate weight vectors, i.e., all components are almost identical, β_i is close to 0, while for boundary weight vectors, particularly those lying on the M coordinate axes, β_i has a value of one. α is a non-negative value controlling the magnitude of penalty and $\alpha = 4.0$ is recommended in this paper based on some preliminary studies. The exponential formulation used here ensures that the penalty value is always positive.

4 Experimental Study

4.1 Experimental Settings

To examine the effectiveness of our proposed schemes, we use six complex problems with irregular Pareto fronts instead of some well-known test suites such as ZDT (Zitzler et al., 2000) or DTLZ (Deb et al, 2005) for empirical comparison. Some of them have been tested in (Jiang and Yang, 2015) and (Deb and Jain, 2014), showing that the original MOEA/D faces difficulties in solving these kinds of problems due to their complex characteristics. The detailed description of these problems is presented in Table 1.

The proposed two penalty schemes, i.e., APS and SPS, are compared with the original PBI. MOEA/D [1] with these different penalty schemes was implemented in C++. θ in the canonical PBI was set to 5.0, and this kind of setting has been widely adopted in the literature (Zhang and Li, 2007; Deb and Jain, 2014; Li et al., 2015a). The neighbourhood size T was set to 20. The crossover probability was set to $p_c = 1.0$ and its distribution index was $\eta_c = 20$. The mutation probability was set to $p_m = 1/n$, where n is the number of variables, and its distribution $\eta_m = 20$. We set the population size N to 100 for bi-objective problems and 190 for three-objective problems. The stopping criterion was set to 100 generations. For all the test problems, the MOEA/D variants were executed 30 independent runs.

4.2 Performance Metrics

In the experiments, we use three performance measures, which are described below.

4.2.1 Maximum Spread (MS)

The MS, first introduced by Zitzler *et al.* (Zitzler et al., 2000), measures to what extent the extreme members in an approximated Pareto front POF^* has been reached. Goh and Tan (Goh and Tan, 2007) proposed a modified version of MS by taking into account the proximity of POF^* towards the true Pareto front POF:

$$MS' = \sqrt{\frac{1}{M}\sum_{k=1}^{M}\left[\frac{min[\overline{POF_k}, \overline{POF_k^*}] - max[\underline{POF_k}, \underline{POF_k^*}]}{\overline{POF_k^*} - \underline{POF_k^*}}\right]^2} \tag{8}$$

where $\overline{POF_k}$ and $\underline{POF_k}$ are the maximum and minimum of the kth objective in POF, respectively; Similarly, $\overline{POF_k^*}$ and $\underline{POF_k^*}$ are the maximum and minimum of the kth objective in POF^*, respectively. A large value of MS' indicates a good spread of POF^*, and MS' will have a value of one when POF^* covers the whole POF.

4.2.2 Inverted Generational Distance (IGD)

The IGD (Zitzler et al., 2003) is an effective performance indicator since it can provide reliable information on both the diversity and convergence of obtained solutions. Let PF be a set of solutions uniformly sampled from the true POF, and PF^* be the approximated solutions in the objective space, the IGD metric measures the gap between PF^* and PF, which is calculated as follows:

$$IGD(PF^*, PF) = \frac{\sum_{p \in PF} d(p, PF^*)}{|PF|} \tag{9}$$

where $d(p, PF^*)$ is the distance between the member p of PF and the nearest member of PF^*. In this paper, 500 points uniformly sampled from the true POF are used as the reference set for IGD, which can be done by generating a large volume of points and then pruning them to the desirable size by the kth nearest neighbour method proposed in the strength Pareto evolutionary algorithm 2 (SPEA2) (Zitzler et al., 2002).

4.2.3 Hypervolume (HV)

HV (Zitzler and Thiele, 1999) measures the size of the objective space dominated by the approximated solutions S and bounded by a reference point $R = (R_1, \ldots, R_M)^T$ that is dominated by all points on the POF, and is computed by:

$$HV(S) = Leb(\bigcup_{x \in S}[f_1(x), R_1] \times \cdots \times [f_M(x), R_M]) \tag{10}$$

where $Leb(A)$ is the Lebesgue measure of a set A. In our experiments, R is set to $(1.2, 1.2)^T$ and $(1.2, 1.2, 1.2)^T$ for bi- and three-objective test instances, respectively.

[1] Source code available from http://dces.essex.ac.uk/staff/qzhang/.

Table 1 Test Instances

Instance	Description	Domain	Number of Variables	Notes
F1	$f_1(x) = (1+g(x))x_1$ $f_2(x) = (1+g(x))(1-\sqrt{x_1})^3$ $g(x) = 2\sin(0.5\pi x_1)(n-1+\sum_{i=2}^{n}(y_i^2-\cos(2\pi y_i)))$ where $y_{i=2:n} = x_i - \sin(0.5\pi x_i)$ POF: $f_2 = (1-\sqrt{f_1})^3$ POS: $x_i = \sin(0.5\pi x_i), i = 2,\ldots,n$	$[0,1]^n$	20	Uni-modal Convex
F2	$f_1(x) = (1+g(x))x_1$ $f_2(x) = (1+g(x))\sqrt{1-{x_1}^5}$ $g(x) = 2\sin(0.5\pi x_1)(n-1+\sum_{i=2}^{n}(y_i^2-\cos(2\pi y_i)))$ where $y_{i=2:n} = x_i - \sin(0.5\pi x_i)$ POF: $f_2 = \sqrt{1-{f_1}^5}$ POS: $x_i = \sin(0.5\pi x_i), i = 2,\ldots,n$	$[0,1]^n$	20	Uni-modal Convex
F3	$f_1(x) = (1+g(x))x_1$ $f_2(x) = \frac{1}{2}(1+g(x))(1-x_1^{0.1}+(1-\sqrt{x_1})^2\cos^2(3\pi x_1))$ $g(x) = 2\sin(0.5\pi x_1)(n-1+\sum_{i=2}^{n}(y_i^2-\cos(2\pi y_i)))$ where $y_{i=2:n} = x_i - \sin(0.5\pi x_i)$ POF: $f_2 = \frac{1}{2}(1-f_1^{0.1}+(1-\sqrt{f_1})^2\cos^2(3\pi f_1))$ POS: $x_i = \sin(0.5\pi x_i), i = 2,\ldots,n$	$[0,1]^n$	20	Multi-modal Disconnected
F4	$f_1(x) = (1+g(x))(x_1+0.05\sin(6\pi x_1))^2$ $f_2(x) = (1+g(x))(1-x_1+0.05\sin(6\pi x_1))^2$ $g(x) = 2\sin(0.5\pi x_1)(n-1+\sum_{i=2}^{n}(y_i^2-\cos(2\pi y_i)))$ where $y_{i=2:n} = x_i - \sin(0.5\pi x_i)$ POF: $f_1^{0.5}+f_2^{0.5} = 1+0.1\sin(3\pi(f_1^{0.5}-f_2^{0.5}+1))$ POS: $x_i = \sin(0.5\pi x_i), i = 2,\ldots,n$	$[0,1]^n$	20	Multi-modal mixed
F5	$f_1(x) = (1+g(x))(x_1+0.05\sin(6\pi x_1))^{0.2}$ $f_2(x) = (1+g(x))(1-x_1+0.05\sin(6\pi x_1))^{10}$ $g(x) = 2\sin(0.5\pi x_1)(n-1+\sum_{i=2}^{n}(y_i^2-\cos(2\pi y_i)))$ where $y_{i=2:n} = x_i - \sin(0.5\pi x_i)$ POF: $f_1^5+f_2^{0.1} = 1+0.1\sin(3\pi(f_1^5-f_2^{0.1}+1))$ POS: $x_i = \sin(0.5\pi x_i), i = 2,\ldots,n$	$[0,1]^n$	20	Multi-modal mixed
F6	$f_1(x) = ((1+g(x))\cos(0.5\pi x_1)\cos(0.5\pi x_2))^4$ $f_2(x) = ((1+g(x))\cos(0.5\pi x_1)\sin(0.5\pi x_2))^4$ $f_3(x) = ((1+g(x))\sin(0.5\pi x_1))^2$ $g(x) = \sum_{i=3}^{n}(x_i-0.5)^2$ POF: $\sqrt{f_1}+\sqrt{f_2}+f_3 = 1$ POS: $x_i = 0.5, i = 3,\ldots,n$	$[0,1]^n$	20	Uni-modal Convex

4.3 Experimental Results

Table 2 records MS, IGD and HV values obtained by PBI, PBI+APS and PBI+SPS for F1 to F6, where the best value for each problem and each performance metric is marked in bold face. From Table 2, it can be seen that both the proposed penalty schemes are helpful for improving the performance of PBI. Specifically, on the MS metric, the use of APS and SPS can provide a better coverage of POF for most of the problems compared with the original PBI, and SPS improves the coverage much more than APS. The IGD metric suggests that APS and SPS can give better distribution of obtained points along the POF for all the problems except F2. Besides, the HV metric further indicates the effectiveness of APS and SPS for most of the problems, and SPS significantly improves the performance of PBI except on F2. Note that, PBI with APS or SPS does not offer noticeable improvement on the MS, IGD and HV metrics for F2. This may be because F2 is a concave problem and there is no considerable difference between boundary solutions and intermediate solutions in terms of their distance to the obtained ideal point. This negligible difference implies that all the decomposed subproblems have similar convergence performance in PBI. In this case, diversity is the only influencing factor, and thus a fixed and identical penalty value can help all the problems reach the same level of balance between diversity and convergence, which may be suitable for solving such kind of problem. Nevertheless, the use of APS and SPS does not degrade PBI's performance in this case.

Table 2 Best, mean and worst performance measure values obtained by three penalty schemes on six problems

	MS			IGD			HV		
Prob.	PBI	PBI+APS	PBI+SPS	PBI	PBI+APS	PBI+SPS	PBI	PBI+APS	PBI+SPS
F1	0.5323	0.6418	**0.9313**	0.1045	0.0665	**0.0181**	1.3137	1.3272	**1.3356**
	0.5252	0.6282	**0.8928**	0.1077	0.0717	**0.0213**	1.3123	1.3256	**1.3354**
	0.5135	0.6012	**0.8129**	0.1125	0.0783	**0.0288**	1.3101	1.3234	**1.3340**
F2	0.9998	0.9993	**0.9999**	**0.0074**	0.0084	0.0083	**0.5404**	0.5402	0.5402
	0.9970	0.9972	**0.9987**	0.0116	**0.0104**	0.0146	0.5388	0.5388	**0.5395**
	0.9845	0.9602	0.9202	0.7838	**0.0391**	0.0537	**0.5334**	0.5265	0.4919
F3	0.5988	0.5994	**0.8852**	0.1941	0.1851	**0.0209**	1.3334	1.3395	**1.3565**
	0.4842	0.5402	**0.8780**	0.4040	0.3966	**0.0224**	1.2974	1.3048	**1.3563**
	0.3290	0.3907	**0.8756**	0.4046	0.3985	**0.0283**	1.2962	1.3031	**1.3553**
F4	0.6192	0.6552	**0.9160**	0.0817	0.0707	**0.0131**	1.2549	1.2576	**1.2656**
	0.6041	0.6297	**0.8933**	0.0838	0.0744	**0.0150**	1.2537	1.2566	**1.2654**
	0.5988	0.5994	**0.8852**	0.0848	0.0791	**0.0165**	1.2527	1.2480	**1.2647**
F5	0.8659	0.9261	**0.9758**	0.0218	0.0109	**0.0092**	0.7530	**0.7573**	0.7565
	0.8561	0.8576	**0.9664**	0.0230	0.0234	**0.0130**	0.7479	0.7449	**0.7540**
	0.6367	0.7617	**0.9245**	0.1391	0.0627	**0.0280**	0.6168	0.6829	**0.7351**
F6	0.7108	0.8441	**1.0000**	0.0425	**0.0329**	0.0337	1.6705	**1.6748**	1.6740
	0.6889	0.7788	**0.9670**	0.0469	**0.0376**	0.0363	1.6644	**1.6705**	1.6702
	0.6676	0.7197	**0.9385**	0.0495	**0.0417**	0.0453	1.6572	**1.6664**	1.6619

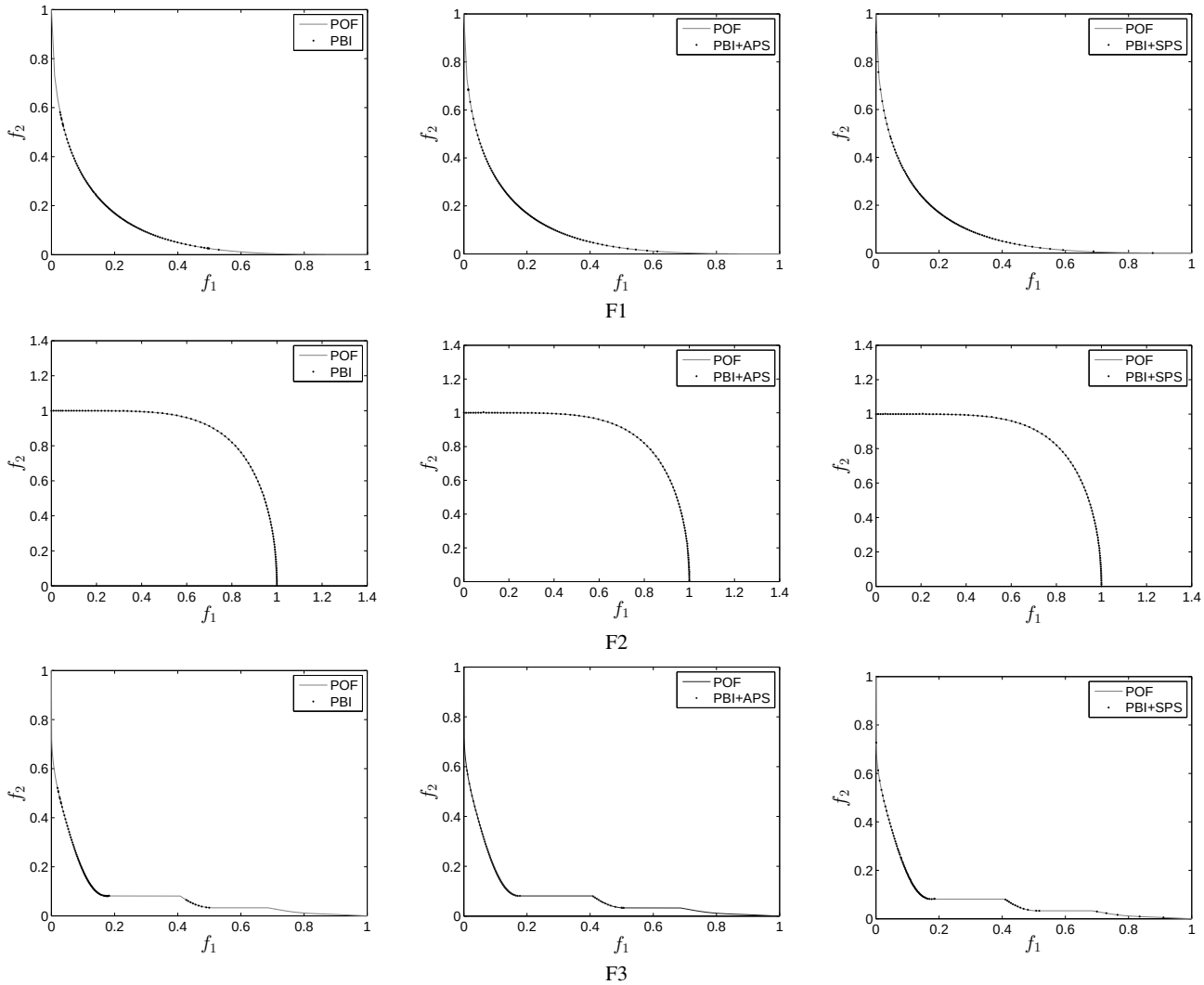


Fig. 6 Approximated POFs with the lowest IGD values among 30 runs on F1-F3.

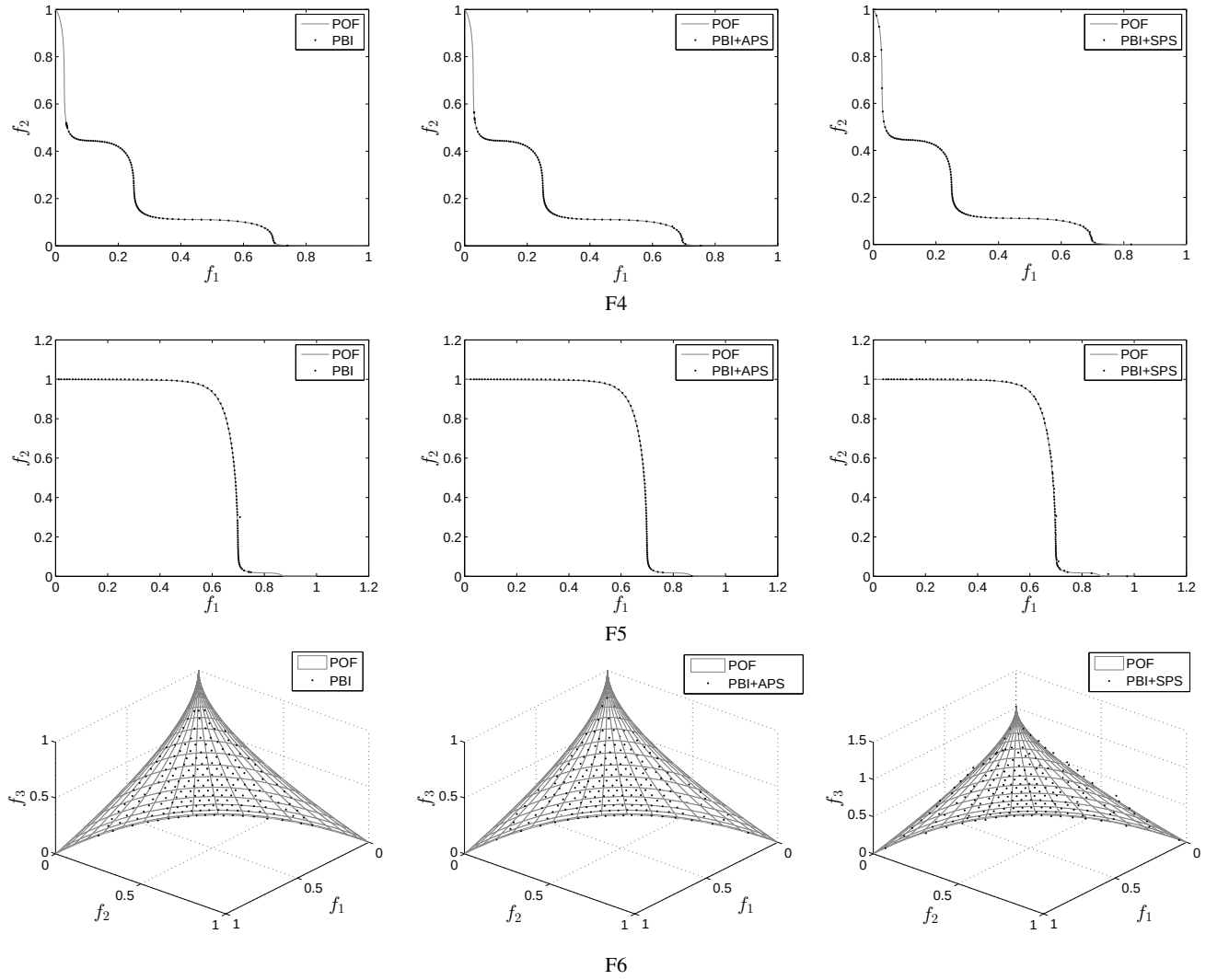


Fig. 7 Approximated POFs with the lowest IGD values among 30 runs on F4-F6.

For a better understanding of the improvement provided by our proposed penalty schemes, we also present graphical plots of approximated POFs for the six problems in Figs. 6 and 7. It can be clearly observed from the figures that, both PBI+APS and PBI+SPS help to find more boundary points than the original PBI for F1, F3, F4 and F6 so as to stretch out their approximated POFs, and PBI+SPS performs much better than PBI+APS in doing this. For the overall concave F2 and F5 problems, three PBI variants show nearly similar distribution of obtained solutions, although PBI+SPS seems to find more boundary points than PBI and PBI+APS for F5.

It is understandable that PBI+SPS performs the best in most cases. This can be attributed to the feature that PBI+SPS imposes strict penalty on boundary subproblems so that it is capable of finding boundary points.

4.4 Integration into MOEA/D Variants

The previous subsection clearly shows the effectiveness of the two proposed penalty schemes, and SPS generally performs better than APS. In this subsection, SPS is integrated into two newly-proposed variants of MOEA/D, i.e., MOEA/D-ACD (Wang et al., 2015) and MOEA/D-STM (Li et al., 2014c), to examine whether SPS can help improve the performance of PBI-based algorithms.

4.4.1 Integration into MOEA/D-ACD

MOEA/D-ACD uses constrained decomposition approaches to convert an MOP into a number of single-objective subproblems. The basic idea behind it is to add constraints to the decomposed subproblems to shrink improvement regions. For the w^i search direction in MOEA/D, the resulting con-

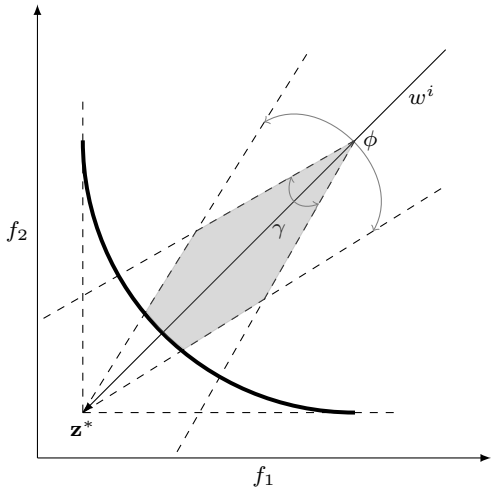


Fig. 8 Illustration of the improvement region (marked in grey) for the constrained PBI approach.

strained subproblem i is defined as follows:

$$\begin{aligned} &\min\ g(\mathbf{x}|\mathbf{w^i}, \mathbf{z}^*) \\ &s.t.\ \langle \mathbf{w^i}, \mathbf{F}(\mathbf{x}) - \mathbf{z}^* \rangle \leq \tfrac{\phi}{2} \\ &\quad x \in \Omega_x \end{aligned} \tag{11}$$

where $g(\cdot)$ is the scalarizing objective function defined by a decomposition approach, $\langle \mathbf{w^i}, \mathbf{F}(\mathbf{x}) - \mathbf{z}^* \rangle$ is the acute angle between $\mathbf{w^i}$ and $\mathbf{F}(\mathbf{x}) - \mathbf{z}^*$, and ϕ is a parameter controlling the size of improvement regions. The value of ϕ is different for different subproblems. Fig. 8 illustrates the improvement region of the constrained PBI approach for subproblem i.

It can be clearly observed from Fig. 8 that the enclosed improvement region for the constrained PBI is mainly controlled by two acute angles, i.e., ϕ and γ, where γ is determined by the penalty factor θ. A constant θ value in MOEA/D-ACD means all subproblems have the same γ, and the improvement regions may be still too large for some subproblems although the constrained decomposition approach is used. As a result, a new solution can replace several different old but well-distributed solutions, thus impairing the population diversity. In contrast, if each subproblem is assigned a different and proper θ value, the improvement regions will be well controlled, which may help maintain the balance between the population diversity and convergence.

It should be noted that our proposed SPS and the constrained approach in MOEA/D-ACD are similar in some sense because both are based on the assumption that each subproblem should separately maintain an appropriately different balance between diversity and convergence. When SPS is integrated into MOEA/D-ACD, it is expected that SPS can help improve the performance of MOEA/D-ACD. For this reason, we compare the integrated MOEA/D with MOEA/D-ACD on our previously used test problems. For notation convenience, MOEA/D-ACD is denoted as "ACD" and MOEA/D-ACD with SPS as "ACD+SPS" later on.

Table 3 Best, mean and worst performance measure values obtained by ACD and ACD+SPS

	MS		IGD		HV	
Prob.	ACD	ACD+SPS	ACD	ACD+SPS	ACD	ACD+SPS
F1	0.8257	**0.9623**	0.0270	**0.0196**	1.3340	**1.3351**
	0.7538	**0.8653**	0.0390	**0.0228**	1.3318	**1.3337**
	0.6466	**0.7405**	0.0640	**0.0440**	1.3267	**1.3311**
F2	**0.9999**	0.9996	**0.0164**	0.0205	**0.5397**	0.5383
	0.9988	0.9982	**0.0239**	0.0320	**0.5363**	0.5342
	0.9914	**0.9936**	0.0399	**0.0371**	**0.5294**	0.5280
F3	0.8343	**0.8816**	0.0278	**0.0190**	1.3531	**1.3559**
	0.7431	**0.8600**	0.0410	**0.0218**	1.3509	**1.3552**
	0.5322	**0.6116**	0.1763	**0.1738**	1.3389	**1.3435**
F4	0.9160	**0.9184**	0.0147	**0.0143**	1.2630	**1.2643**
	0.8600	**0.8930**	0.0208	**0.0166**	1.2613	**1.2626**
	0.8020	**0.8723**	0.0274	**0.0200**	1.2584	**1.2606**
F5	0.9865	**0.9889**	**0.0085**	0.0105	0.7548	**0.7578**
	0.9210	**0.9278**	0.0173	**0.0137**	0.7497	**0.7502**
	0.8659	**0.8937**	0.0265	**0.0257**	**0.7420**	0.7419
F6	0.9202	**1.0000**	0.0358	**0.0310**	1.6713	**1.6748**
	0.7623	**0.9986**	0.0379	**0.0331**	1.6662	**1.6718**
	0.7197	**0.9829**	0.0408	**0.0388**	1.6569	**1.6686**

Table 4 Statistical difference between ACD and ACD+SPS

	MS		IGD		HV	
Sign	ACD	ACD+SPS	ACD	ACD+SPS	ACD	ACD+SPS
better	0	5	0	4	1	4
equivalent	1	1	2	2	1	1
worse	5	0	4	0	4	1

The results of ACD and ACD+SPS regarding the three performance measures are given in Table 3. From Table 3, we can observe that the use of SPS helps enhance the performance of ACD for all the problems except F2 and F5. For F2 and F5, ACD performs better in some runs but worse in other runs than ACD+SPS. The comparison between ACD and ACD+SPS clearly indicates that different subproblems require different θ values to control the upper bound of the improvement regions. Furthermore, we performed the Wilcoxon rank sum test on these two algorithms at the 0.05 significant level to verify the statistical difference between them. The results are shown in Table 4, where "better", "equivalent", or "worse" denotes the number of test problems on which the corresponding algorithm is better than, equivalent to, or worse than the compared algorithm, respectively. The statistical results further confirm that SPS is beneficial to MOEA/D and improves the performance of MOEA/D-ACD.

In order to have a better understanding of the impact of SPS, we graphically plot the average IGD metric obtained by ACD with and without SPS against the number of generations on the six test problems in Fig. 9. Clearly, ACD without SPS generally has a faster convergence speed than that with SPS. The reason is that the use of SPS reduces the improvement region for each subproblem so that ACD+SPS needs some effort to find a new solution that is better, in

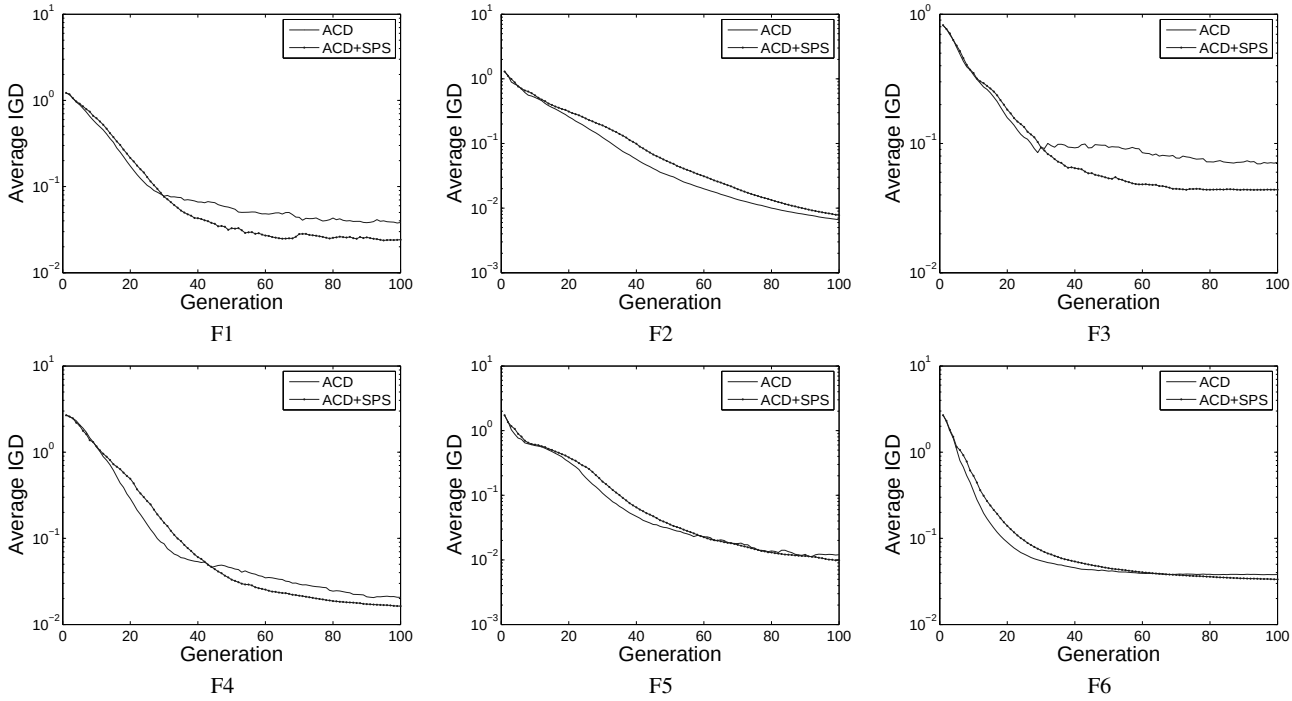


Fig. 9 Evolution curves of the average IGD metric for F1-F6.

terms of diversity and convergence, than the old solution of the considered subproblem and replace the old with the new one. Despite that, the use of SPS improves the IGD metric at the late stage for the problems except F2. This means that SPS enhances the diversity performance of MOEA/D-ACD, thereby providing improvement on the distribution of obtained POFs.

4.4.2 Integration into MOEA/D-STM

Another MOEA/D variant to be considered is a stable matching (STM) based MOEA, called MOEA/D-STM (Li et al., 2014c). MOEA/D-STM uses the STM model to coordinate the selection process in MOEA/D, where subproblem agents can express their preferences over the solution agents, and vice versa. As a consequence, suproblems' preference encourages convergence whereas solutions' preference promotes diversity. As our work focuses on the improvement of PBI, so PBI should be used in MOEA/D-STM for experimental analysis, and the penalty value θ is set to 5.0 (Zhang and Li, 2007). The other parameter settings remain the same as in Li et al. (2014c).

Table 5 reports the results of three performance measures obtained by MOEA/D-STM and MOEA/D-STM with SPS, where for the notation convenience, "STM" and "STM+SPS" represent the former and the latter, respectively. It can be clearly observed that, the use of SPS significantly improves the performance of STM for all the considered problems. On the other hand, it is worth noting that STM seems struggling to solve F2, and its MS metric value of zero indicates that STM produces as many duplicate solutions as the population size for F2. In contrast, when SPS is integrated into STM, F2 can be easily approximated, as indicated by the very small IGD values. This means a fixed penalty value for PBI in STM is not suitable for solving test problems.

The poor performance of STM without SPS on the tested problems may be explained by two possible reasons. One is

Table 5 Best, mean and worst performance measure values obtained by STM and STM+SPS

	MS		IGD		HV	
Prob.	STM	STM+SPS	ACD	STM+SPS	STM	STM+SPS
F1	0.5375	**0.8960**	0.1025	**0.0188**	1.3139	**1.3351**
	0.5273	**0.8676**	0.1068	**0.0221**	1.3123	**1.3346**
	0.5130	**0.8186**	0.1124	**0.0338**	1.3099	**1.3326**
F2	0.0000	**1.0000**	0.7838	**0.0159**	0.2400	**0.5399**
	0.0000	**1.0000**	0.7838	**0.0219**	0.2400	**0.5349**
	0.0000	**1.0000**	0.7838	**0.0407**	0.2400	**0.5283**
F3	0.4796	**0.8822**	0.1978	**0.0211**	1.3319	**1.3560**
	0.3311	**0.8770**	0.4036	**0.0252**	1.2976	**1.3547**
	0.3191	**0.8407**	0.4051	**0.0331**	1.2956	**1.3526**
F4	0.6579	**0.9248**	0.0805	**0.0138**	1.2538	**1.2645**
	0.6066	**0.8949**	0.0846	**0.0166**	1.2525	**1.2637**
	0.5998	**0.8857**	0.0861	**0.0194**	1.2519	**1.2624**
F5	0.8784	**0.9817**	0.0253	**0.0114**	0.7546	**0.7618**
	0.8719	**0.9478**	0.0313	**0.0154**	0.7533	**0.7598**
	0.8678	**0.9370**	0.0406	**0.0206**	0.7517	**0.7578**
F6	0.7373	**1.0000**	0.0389	**0.0334**	1.6693	**1.6725**
	0.7094	**1.0000**	0.0421	**0.0355**	1.6664	**1.6714**
	0.6847	**0.9943**	0.0444	**0.0378**	1.6625	**1.6696**

Table 6 Best, median and worst IGD values obtained by PBI+SPS with different α values on F1 and F2

Prob.	$\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$	$\alpha = 4.0$	$\alpha = 5.0$	$\alpha = 10.0$
F1	0.1757	0.0791	0.0351	0.0181	0.0186	0.0193
	0.1761	0.0825	0.0391	0.0213	0.0203	0.0219
	0.1771	0.0877	0.0575	0.0288	0.0325	0.0287
F2	0.0047	0.0047	0.0047	0.0047	0.0048	0.0090
	0.0048	0.0048	0.0048	0.0049	0.0052	0.0137
	0.7835	0.7835	0.0071	0.0051	0.0067	0.0230

that, the test problems may be too complicated in terms of their POF shapes so that STM struggles to solve them. The other reason may be the inappropriate setting of the penalty value in PBI. As subproblems in MOEA/D are responsible for different landscapes of the objective space, they should be given different penalty values for different scenarios to balance convergence and diversity in PBI. Although STM promotes the balance between convergence and diversity, its convergence still depends on scalarizing functions. Thus, when PBI is chosen as the scalarizing function, the penalty value should be carefully specified.

4.5 Influence of Parameter α on SPS

In SPS, the magnitude of penalty for a subproblem considerably depends on its control parameter α. In this subsection, we investigate the sensitivity of SPS to this parameter. MOEA/D with PBI+SPS is tested on F1 and F2 since these two problems have distinctly different POF shapes, i.e., F1 is extremely convex whereas F2 is extremely concave. The value of α in SPS varies from 1.0 to 10.0, and other parameters remain the same as in Section 4.1.

Table 6 presents the obtained IGD values of MOEA/D with PBI+SPS with different α values on F1 and F2. It can be observed that, for F1, $\alpha = 4.0$ in SPS provides better IGD values than the other settings. Setting α to a too small or too large value degrades the performance of PBI+SPS in terms of the IGD metric. For F2, however, all the settings of α except $\alpha = 10.0$ can produce similar IGD results and there is no significant difference between these IGD values. This observation indirectly indicates that, concave problems are less likely to be affected by the penalty factor θ in PBI, but this is not the case with convex problems.

In SPS, the penalty value is controlled by two parameters, i.e., α and β. For a weight vector w^i, β_i can be easily calculated by Eq. (6). Thus, α is the only factor that can influence the penalty value. However, for distinct weight vectors, the value of $\alpha\beta$ may be different. There are at least two scenarios, i.e., $\alpha\beta < 1$ or $\alpha\beta \geq 1$. Small $\alpha\beta$ values underemphasize population diversity whereas large ones favour diversity. To investigate these situations, we use F1 as the test problem. As F1 has an extremely convex POF shape, the boundary regions are hard to be located. Therefore, we

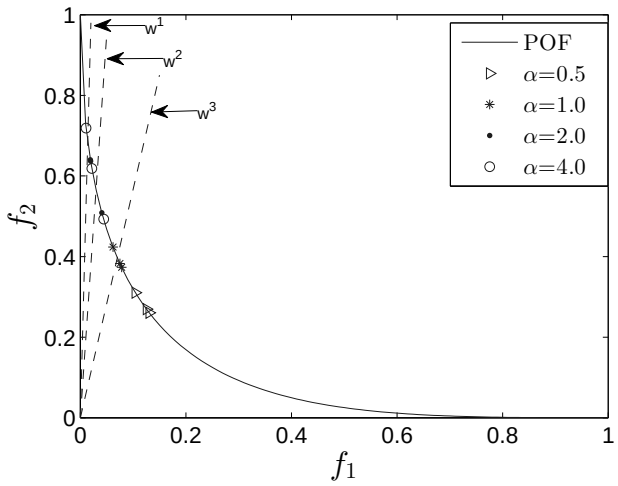


Fig. 10 Influence of α on population distribution for F1.

select three boundary weight vectors $w^1 = (0.02, 0.98)^T$, $w^2 = (0.05, 0.95)^T$ and $w^3 = (0.15, 0.85)^T$ to approximate three boundary points. Correspondingly, their β values are $\beta_1 = 0.96$, $\beta_2 = 0.9$, and $\beta_3 = 0.7$, respectively. α is set as α=0.5, 1, 2, and 4, meaning that $\alpha\beta < 1$ if α=0.5 and 1; otherwise, $\alpha\beta > 1$. Since only three population individuals (which is determined by the number of weight vectors) are used, we increase the maximum number of generations to 1000 in order to guarantee convergence.

Figure 10 shows the influence of α on population distribution. Clearly, the larger α is, the more spread the obtained points will be. Thus, $\alpha = 4$ yields better spread than the other settings, although one of the solutions is a little bit far away from w^3. We guess that $\alpha = 4$ might be too large for the subproblem associated with w^3, which may discard solutions that have good convergence but poor diversity. On the other hand, small α values favour intermediate points as these points are much closer than boundary points to the ideal point, indicating that the resulting penalty is not enough to maintain diversity.

It is worth noting that $\alpha = 4.0$ might not be the best parameter setting for all kinds of problems, although it clearly provides a significant improvement for PBI. As α controls the balance between the population diversity and convergence for each decomposed subproblem, it should be carefully selected in order to maximize the algorithm's performance. Also, it is desirable to design an adaptive strategy for the setting of α, which is left for our future work.

5 Conclusions and Future Work

In this paper we have investigated the influence of the penalty factor on the performance of PBI within MOEA/D. The investigation shows that, a small penalty value is bene-

ficial for convergence but may make PBI unable to preserve boundary solutions if boundary solutions are much further away from the obtained ideal point than intermediate solutions, while a large one is advantageous to diversity but can slow down the convergence course. On the basis of this investigation, two new penalty schemes, i.e., APS and SPS, are introduced to strike a balance between convergence and diversity. Similar to the original PBI, APS assigns identical penalty values for all subproblems, but the penalty values are generationally increasing during the course of search, with a small value at the early stage in favour of fast convergence and a large one at the late stage emphasizing diversity and spread of solutions. In contrast, SPS turns to specify distinct penalty values for different subproblems according to the location of their associated weight vectors. Boundary subproblems are given a strict penalty and intermediate ones are given a loose penalty. This way, preservation of boundary solutions are enhanced so as to provide a wide spread of Pareto font.

These two new penalty schemes have been tested on several complex problems, including bi-objective and three-objective problems. Experimental studies have shown that both schemes help ease the loss of boundary solutions and outspread the Pareto font, and PBI with SPS performs significantly better than that with APS. This means SPS is an effective way to improve the performance of PBI. Besides, SPS has been integrated into two recently proposed variants of MOEA/D, i.e., MOEA/D-ACD and MOEA/D-STM, and experimental results have clearly demonstrated the efficiency of SPS in promoting the performance of decomposition-based MOEAs with PBI.

Although the proposed penalty schemes have offered encouraging results on the test problems considered in this paper, they should be examined on a wider range of different kinds of problems. As there has recently been an increasing number of studies in extending PBI-based MOEAs to many-objective optimization, it should be very interesting to further examine the effect of PBI on different types of many-objective problems. Besides, designing a parameter-free or adaptive PBI method for MOEA/D is very desirable. In our future work, MOEA/D with PBI variants will be compared with other state-of-the-art algorithms.

Acknowledgement

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) of U.K. under Grant EP/K001310/1 and the National Natural Science Foundation of China under Grant 61273031.

References

Al Moubayed N, Petrovski A, McCall J (2013) D^2MOPSO: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces. Evol Comput 22(1):47–77

Asafuddoula M, Ray T, Sarker R (2015) A decomposition based evolutionary algorithm for many objective optimization. IEEE Trans Evol Comput 19(3):445–460

Beume N, Naujoks N, Emmerich M (2007) SMS-EMOA: Multiobjective selection based on dominated hypervolume. Eur J Oper Res 181(3):1653–1669

Bader J, Zitzler E (2011) HypE: An algorithm for fast hypervolume-based many-objective optimization. Evol Comput 19(1):45–76

Cheng R, Jin Y, Narukawa K (2015) Adaptive reference vector generation for inverse model based evolutionary multi-objective optimization with degenerate and disconnected Pareto fronts. In: Evolutionary Multi-Criterion Optimization (EMO 2015), Part I, LNCS 9018, pp 127–140

Cheng R, Jin Y, Olhofer M, Sendhoff B (2016). A reference vector guided evolutionary algorithm for many-objective optimization. IEEE Trans Evol Comput (in press), doi: 10.1109/TEVC.2016.2519378

Chikumbo O, Goodman ED, Deb K (2012) Approximating a multi-dimensional pareto front for a land use management problem: A modified moea with an epigenetic silencing metaphor. In: Proceedings of 2012 IEEE Congress on Evolutionary Computation (CEC 2012), pp 1–9

Deb K, Agrawwal S, Pratap A, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197

Deb K, Jain H (2014) An evolutionary many-Objective optimization algorithm using reference-point based non-dominated sorting approach, Part I: Solving problems with box constraints. IEEE Trans Evol Comput 18(4):577–601

Deb K, Pratap A, Moitra S (2000) Mechanical component design for multiple objectives using elitist non-dominated sorting GA. In: Proceedings of the 6th International Conference on Parallel Problem Solving from Nature (PPSN VI), pp 859–868

Deb K, Thiele L, Laumanns M, Zitzler E (2005) Scable test problems for evolutionary multi-objective optimization. In: Abraham A, Jain L, Goldberg R (Eds.) Evolutionary Multiobjective Optimization: Theoretical Advances and Applications, Springer, London, pp 105–145

Giagkiozis I, Purshouse RC, Fleming PJ (2014) Generalized decomposition and cross entropy methods for many-objective optimization. Inform Sci 282:363–387

Goh C, Tan KC (2007) An investigation on noisy environments in evolutionary multiobjective optimization. IEEE Trans Evol Comput 11(3):354–381

Gomez RH, Coello Coello CA (2015) Improved metaheuristic based on the r2 indicator for many-objective optimization. In: Proceedings of the 2015 Genetic and Evolutionary Computation Conference (GECCO 15), pp 679–686

Huband S, Hingston P, Barone L, While L (2006) A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans Evol Comput 10(2):477–506

Ishibuchi H, Akedo N, Nojima Y (2015) Behavior of multiobjective evolutionary algorithms on many-objective knapsack problems. IEEE Trans Evol Comput 19(2):264–283

Ishibuchi H, Akedo N, Nojima Y (2013) A study on the specification of a scalarizing function in MOEA/D for many-objective knapsack problems. In: Proceedings of the 7th International Conference on Learning and Intelligent Optimization 7 (LION 7), LNCS 7997, pp 231–246

Ishibuchi H, Tsukamoto N, Hitotsuyanagi Y, Nojima Y (2010) Indicator-based evolutionary algorithm with hypervolume approximation by achievement scalarizing function. In: Proceedings of 12th Annual Conference on Genetic Evolutionary Computation (GECCO 2010), pp 527–534

Jain H, Deb K (2014) An improved adaptive approach for elitist nondominated sorting genetic algorithm for many-objective optimization. In: Evolutionary Multi-Criterion Optimization (EMO 2013), LNCS 7811, pp 307–321

Jia S, Zhu J, Du B, Yue H (2011) Indicator-based particle swarm optimization with local search. In: Proceedings of the 7th International Conference on Natural Computation, pp 1180–1184

Jiang S, Yang S (2016) An improved multi-objective optimization evolutionary algorithm based on decomposition for complex Pareto fronts. IEEE Trans Cybern. 46(2):421-437

Knowles JD, Corne DW (1999) The pareto archived evolution strategy: A new baseline algorithm for multiobjective optimisation. In: Proceedings of the 1999 IEEE Congress on Evolutionary Computation (CEC 1999), pp 98–105

Li B, Li J, Tang K, Yao X (2014a) An improved two archive algorithm for many-objective optimization. In: Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC 2014), pp 2869–2876

Li K, Deb K, Zhang Q, Kwong S (2015a) An evolutionary many-Objective optimization algorithm based on dominance and decomposition. IEEE Trans Evol Comput 19(5):694–716

Li K, Fialho A, Kwong S, Zhang Q (2014b) Adaptive operator selection with bandits for multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 18(1):114–130

Li K, Kwong S, Zhang Q, Deb K (2015b) Interrelationship-based selection for decomposition multiobjective optimization. IEEE Trans Cybern 45(10):2076–2088

Li K, Zhang Q, Kwong S, Li M, Wang R (2014c) Stable matching based selection in evolutionary multiobjective optimization. IEEE Trans Evol Comput 18(6):909–923

Li H, Zhang Q (2009) Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II. IEEE Trans Evol Comput 13(2):284–302

Liu H, Gu F, Zhang Q (2014) Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. IEEE Trans Evol Comput 18(3):450–455

Masoomi Z, Mesgari MS, Hamrah M (2013) Allocation of urban land uses by multi-objective particle swarm optimization algorithm. Int J Geogr Inf Sci 27(3):542–565

Mendez AM, Coello Coello CA (2015) GD-MOEA: A new multi-objective evolutionary algorithm on the generation distance indicator, In: Evolutionary Multi-Criterion Optimization, LNCS 9018, pp 156–170

Mohammadi A, Omidvar MN, Li X, Deb K (2014) Integrating user preferences and decomposition methods for many-objective optimization. In: Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC 2014), pp 421–428

Osycza A, Kundu S (1995) A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. Struct Optimization 10(2):94–99

Pavelski LM, Delgado MR, de Almeida CP, Goncalves RA, Venske SM (2014) ELMOEA/D-DE: Extreme learning surrogate models in multi-objective optimization based on decomposition and differential evolution. In: Proceedings of the 2014 Brazilian Conference on Intelligent Systems (BRACIS), pp 318–323

Qi T, Ma X, Liu F, Jiao L, Sun J, Wu J (2014) MOEA/D with adaptive weight adjustment. Evol Comput 22(2):231–264

Reed PM, Hadka D, Herman JD, Kasprzyk JR, Kollat JB (2013) Evolutionary multiobjective in water resources: The past, present, and future. Adv Water Resour 51:438–456

Sato H (2014) Inverted PBI in MOEA/D and its impact on the search performance on multi and many-objective optimization. In: Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO 14), pp 645–652

Wang G, Chen J, Cai T, Xin B (2013) Decomposition-based multi-objective differential evolution particle swarm optimization for the design of tubular permanent magnet linear synchronous motor. Eng Optimiz 45(9):1107–1127

Wang L, Zhang Q, Zhou A, Gong M, Jiao L (2015) Constrained subproblems in decomposition based multiobjective evolutionary algorithm. IEEE Trans Evol Comput (in press), doi:10.1109/TEVC.2015.2457616

Yang XS, Deb S (2013) Multiobjective cuckoo search for design optimization. Comput Oper Res 40(6):1616–1624

Yuan Y, Xu H, Wang B (2014) Evolutionary many-objective optimization using ensemble fitness ranking. In: Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO 2014), pp 669–676

Yuan Y, Xu H, Wang B, Yao X (2015) A new dominance relation based evolutionary algorithm for many-objective optimization. IEEE Trans Evol Comput. 20(1):16–37

Zhang Q and Li H (2007) MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 11(6):712–731

Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: Empirical results. Evol Comput 8(2):173–195

Zitzler E, Kunzli S (2004) Indicator-based selection in multiobjective search. In: Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII), vol 3242, pp 832–842

Zitzler E, Laumanns M, Thiele L (2002) SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: Proceedings of Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001), vol 3242, no 103, pp 95–100

Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Trans Evol Comput 3(4):257–271

Zitzler E, Thiele L, Laumanns M, Fonseca CM, da Fonseca VG (2003) Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans Evol Comput 7(2):117–132